

## FR2667419

Publication Title:

Memory-card application-program debugging process and debugging system

Abstract:

The invention relates to systems for debugging chip card (smart card) application programs. It applies to cards including a microprocessor and an electrically programmable non-volatile memory, this memory containing an application program which can be run by the microprocessor.

In order to carry out debugging, a very simple system is used which relies on the fact that the memory of the application program is electrically programmable. This memory is used to store, on the one hand, provisional and modifiable versions of the application program and, on the other hand, a debugging aid program. In practice the debugging system comprises merely a specimen chip card (30), a card reader (32) and a microcomputer (34) for controlling the exchanges between the card and the reader. The process consists essentially in storing in memory the program to be debugged, in modifying an instruction of this program in order to replace it with a branch instruction to the debugging aid program, and in initiating the application program. The aid program provides information regarding the status of the system at the time of the branching.

-----  
Data supplied from the esp@cenet database - <http://ep.espacenet.com>

⑨ RÉPUBLIQUE FRANÇAISE  
INSTITUT NATIONAL  
DE LA PROPRIÉTÉ INDUSTRIELLE  
PARIS

⑪ N° de publication :  
(à n'utiliser que pour les  
commandes de reproduction)

2 667 419

⑫ N° d'enregistrement national :

90 12114

⑮ Int Cl<sup>8</sup> : G 06 K 19/073

⑫

## DEMANDE DE BREVET D'INVENTION

A1

⑫ Date de dépôt : 02.10.90.

⑬ Priorité :

⑦ Demandeur(s) : Société dite GEMPLUS CARD  
INTERNATIONAL Société Anonyme — FR.

⑧ Inventeur(s) : Sourenian Paul et Geronimi François.

⑭ Date de la mise à disposition du public de la  
demande : 03.04.92 Bulletin 92/14.

⑯ Liste des documents cités dans le rapport de  
recherche : Se reporter à la fin du présent fascicule.

⑰ Références à d'autres documents nationaux  
apparentés :

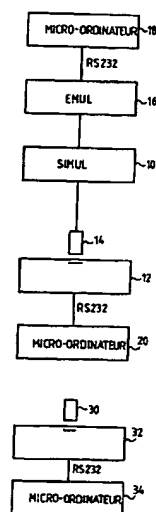
⑨ Titulaire(s) :

⑩ Mandataire : Cabinet Ballot-Schmit.

⑪ Procédé de débogage de programme d'application de carte à mémoire et système de débogage.

⑫ L'invention concerne les systèmes de débogage de programmes d'application de cartes à puces. Elle s'applique aux cartes comportant un microprocesseur et une mémoire non volatile programmable électriquement, cette mémoire contenant un programme d'application exécutable par le microprocesseur.

Pour réaliser le débogage on utilise un système très simple fondé sur le fait que la mémoire du programme d'application est programmable électriquement. On se sert de cette mémoire pour stocker d'une part des versions provisoires et modifiables du programme d'application, et d'autre part un programme d'aide au débogage. Le système de débogage comprend en pratique seulement une carte à puce échantillon (30), un lecteur de carte (32), et un micro-ordinateur (34) pour piloter les échanges entre la carte et le lecteur. Le procédé consiste essentiellement à stocker en mémoire le programme à déboguer, à modifier une instruction de ce programme pour la remplacer par une instruction de branchement vers le programme d'aide au débogage, et à lancer le programme d'application. Le programme d'aide fournit des renseignements sur l'état du système au moment du branchement.



FR 2 667 419 - A1



1

PROCEDE DE DEBOGAGE DE PROGRAMME D'APPLICATION  
DE CARTE A MEMOIRE ET SYSTEME DE DEBOGAGE

L'invention concerne la mise au point de programmes d'application de cartes à puces incorporant un microprocesseur et une mémoire de programmes.

Si la carte à puce comprend un microprocesseur, c'est pour pouvoir exécuter des programmes d'application qui se présentent sous forme d'instructions successives données au microprocesseur. Ces instructions portent sur des opérations internes à la carte et sur les échanges de données effectués entre la carte et un lecteur de carte dans lequel la carte est insérée.

Pour la mise au point de tout programme d'application de microprocesseur, il est nécessaire de passer par une phase de débogage dans laquelle le programme d'application est testé, dans des conditions aussi proches que possible de la réalité, et les erreurs éventuelles sont détectées puis corrigées. Les cartes à puces à microprocesseur n'échappent pas à cette règle.

Les cartes à puces qui sont plus spécialement concernées par la présente invention sont les cartes à microprocesseur comportant non seulement une mémoire morte (ROM) et une mémoire vive de travail (RAM), mais aussi une mémoire électriquement effaçable et reprogrammable (EEPROM) ou simplement électriquement programmable (EPROM). La mémoire morte comprend un programme figé, intangible, représentant notamment le système d'exploitation de la carte à microprocesseur : gestion des mémoires, gestion des sécurités d'accès, et plus généralement tous les programmes obligatoires non modifiables de la carte à puce. La mémoire RAM sert classiquement à stocker des données temporairement et de

manière volatile au cours de l'exécution d'un programme. La mémoire EEPROM ou EPROM contient des données non volatiles et peut contenir aussi des programmes d'application variés, pour que la carte puisse avoir des  
5 fonctionnalités supplémentaires, spécifiques d'une application donnée.

L'invention s'intéresse tout particulièrement au débogage des programmes d'application stockés en mémoire non volatile EEPROM ou EPROM et exécutables directement  
10 à partir de cette mémoire.

Dans la suite on ne parlera que de mémoires EEPROM, c'est-à-dire non seulement programmables mais aussi effaçables électriquement, l'invention étant particulièrement intéressante dans ce cas.

15 Pour effectuer le débogage d'un programme d'application qui est ainsi destiné à être stocké en mémoire non volatile pour être exécuté, on utilise habituellement un outil de développement qui comprend d'une part un simulateur de carte et d'autre part un  
20 émulateur de microprocesseur.

La figure 1 rappelle la constitution classique d'un outil de développement d'application de microprocesseur.

Le simulateur de carte 10 est un appareil électronique destiné à être connecté d'une part à un  
25 lecteur de carte 12 (par l'intermédiaire d'une extension 14 simulant les contacts d'accès d'une carte à puce), et d'autre part à l'émulateur 16. Le simulateur remplace la carte et possède toutes les ressources de la carte (mémoires RAM, ROM, EEPROM, interfaces d'entrée/sortie  
30 etc.) sauf le microprocesseur.

L'émulateur 16 simule le fonctionnement du microprocesseur de la carte et comprend à cet effet un microprocesseur identique à celui des cartes utilisées dans l'application. L'émulateur est relié d'une part au

simulateur de carte 10 pour que son microprocesseur soit dans le même environnement que s'il était monté effectivement sur une carte avec les ressources de celle-ci; et il est relié d'autre part à un  
5 microordinateur 18 capable de le commander c'est-à-dire de lui fournir toute instruction souhaitée et capable d'échanger des données avec lui.

Le lecteur de carte 12 quant à lui est relié aussi à un autre microordinateur 20 capable de le commander  
10 pour lui permettre d'échanger des données avec le simulateur de carte par l'intermédiaire de l'extension 14, exactement comme si une véritable carte était insérée dans le lecteur. C'est par ce deuxième microordinateur 20 que l'on pourra recueillir toutes les  
15 informations utiles sur ce qui se passe dans la carte, le premier microordinateur ayant pour rôle de fournir des instructions au microprocesseur pour lui faire exécuter des programmes contrôlés.

Le développement de l'application avec cet outil  
20 classique consiste à :

- faire exécuter un programme d'application par le microprocesseur de l'émulateur, soit complètement soit partiellement, soit en mode pas-à-pas, soit en imposant des points d'arrêts, etc.
- 25 - examiner le fonctionnement global et détaillé, en s'intéressant notamment aux contenus des mémoires et registres internes du microprocesseur à diverses étapes du programme.
- détecter les défauts de fonctionnalités et  
30 autres erreurs;
- modifier le programme d'application dans un sens tendant à supprimer les défauts constatés;
- et recommencer le débogage avec le programme d'application modifié, ceci jusqu'à disparition complète

des défauts constatés.

Il faut donc un outil de développement relativement compliqué (un émulateur + un simulateur + deux ordinateurs personnels) et donc cher pour effectuer ce débogage.

D'autre part le simulateur de carte est rarement parfaitement compatible avec la carte qu'il doit simuler : il peut y avoir des écarts de conditions d'opération (tension d'alimentation, fréquence d'horloge, jeu d'instructions, etc. ). En effet, on ne modifie pas les outils de débogage, émulateur et simulateur, chaque fois qu'on apporte des petites modifications aux séries de cartes fabriquées.

Enfin, l'utilisateur de la carte, qui développe l'application, n'est pas le fabricant de la carte et du matériel de simulation et il peut être à l'autre bout de la planète. La maintenance de l'outil de développement peut donc poser des problèmes difficiles.

Un but de l'invention est d'améliorer les outils de débogage pour les cartes à puces à microprocesseur incorporant une mémoire non volatile programmable électriquement, susceptible de contenir un programme d'application.

Selon l'invention, on propose un système de débogage de programme d'application de carte à mémoire comprenant une carte à mémoire échantillon correspondant à celle utilisée dans l'application à déboguer, cette carte comportant au moins un microprocesseur et une mémoire non volatile programmable électriquement, un lecteur de carte dans lequel cette carte est introduite, un microordinateur (ordinateur personnel ou autre console de contrôle) pour contrôler le lecteur et notamment les échanges de données entre la carte et le lecteur, le programme à déboguer étant contenu dans une

première zone de la mémoire non volatile de la carte, et un programme d'aide au débogage étant contenu dans une deuxième zone de la mémoire non volatile.

En pratique, le programme d'aide au débogage  
5 comporte des moyens pour sauvegarder dans la mémoire non volatile des données internes représentant le fonctionnement de la carte à un instant donné.

On peut notamment charger dans la mémoire non volatile un programme d'application comportant une  
10 instruction modifiée à l'endroit où on veut observer le comportement interne de la carte, cette instruction modifiée étant une instruction de branchement vers le programme d'aide au débogage.

Le procédé de débogage selon l'invention comprend  
15 les opérations consistant à :

- introduire une carte échantillon dans un lecteur de cartes contrôlé par un microordinateur, la carte comportant un microprocesseur et une mémoire non volatile programmable électriquement,
- 20 - charger dans la mémoire non volatile, à l'aide du microordinateur et du lecteur, un programme d'application à déboguer, et un programme d'aide au débogage,
- donner par l'intermédiaire du  
25 microordinateur et du lecteur des ordres de modification et/ou d'exécution du programme chargé, l'exécution étant éventuellement modifiée sous le contrôle du programme d'aide au débogage, et recueillir dans le microordinateur les données résultant de l'exécution du  
30 programme,
- modifier le programme d'application s'il y a lieu en fonction des résultats et recommencer les étapes de chargement d'un programme d'application, d'exécution, et de recueil des données.

Le programme d'aide au débogage pourra comporter des moyens pour simuler la suite d'une séquence d'échanges entre la carte et le microprocesseur lorsqu'une telle séquence a été interrompue en vue de la  
5 sauvegarde du contexte machine dans la mémoire non volatile.

Par conséquent, au lieu d'utiliser un simulateur de carte qui comprend toutes les ressources de la carte sauf le microprocesseur, un émulateur qui comprend le  
10 microprocesseur, et un microordinateur qui comprend un programme pour le microprocesseur, on utilise simplement une carte échantillon, identique à celle qui sera effectivement utilisée dans l'application. Cette carte comprend ses propres ressources (mémoires vives, mortes  
15 et non volatiles), son microprocesseur, un programme d'application stocké en mémoire non volatile et un programme d'aide au débogage stocké dans la même mémoire et apte à modifier partiellement l'exécution du programme d'application. Seul un lecteur de carte et un  
20 seul microordinateur pour le contrôler sont nécessaires pour effectuer le débogage. Le débogage s'effectue par chargements successifs de programmes modifiés jusqu'à élimination complète des erreurs. On utilise ainsi très  
avantageusement le fait que le programme d'application  
25 est directement exécutable à partir d'une mémoire non volatile et le fait que cette mémoire est programmable électriquement à partir du lecteur de cartes.

Le microordinateur est capable notamment d'insérer des points d'arrêt (instructions de branchement vers le  
30 programme d'aide au débogage) à des points désignés du programme d'application. Le programme d'aide au débogage stocké en mémoire non volatile est de préférence capable de lire et modifier les contenus des diverses zones de mémoire (vive et non volatile) et de sauvegarder en



mémoire non volatile les contenus de registres divers de la carte, cela à n'importe quel moment du processus d'échange entre la carte et le lecteur.

D'autres caractéristiques et avantages de l'invention apparaîtront à la lecture de la description détaillée qui suit et qui est faite en référence aux dessins dans lesquels :

- la figure 1, déjà décrite, représente la constitution classique d'un outil de développement d'application de carte à puce à microprocesseur;

- la figure 2 représente la constitution de l'outil de développement selon l'invention;

- la figure 3 représente schématiquement le contenu de la puce d'une carte à puce;

- la figure 4 représente l'organisation de la mémoire non volatile avec des zones de données et des zones de codes exécutables;

- la figure 5 représente un exemple de séquence d'échanges entre une carte à puce et un lecteur de carte.

La figure 2 représente l'outil de développement selon l'invention permettant le débogage d'un programme d'application d'une carte à microprocesseur lorsque ce programme est stocké dans une mémoire non volatile interne, programmable électriquement (en général EEPROM donc également effaçable électriquement), de la puce de circuit intégré de la carte.

L'outil est très simple : il comprend

- une carte échantillon 30 correspondant exactement aux cartes qui seront utilisées lorsque le programme sera définitivement au point;

- un lecteur de carte 32 dans lequel on peut insérer la carte et échanger avec elle des données selon une procédure qui n'est pas différente de la procédure qu'on utilisera dans l'application finale;

- et un microordinateur 34 pour contrôler le lecteur et donc les échanges avec la carte. Les échanges avec la carte sont classiquement des échanges en série. Les échanges entre le microordinateur et le lecteur  
5 peuvent se faire par une liaison classique type RS232.

La carte échantillon, comme la carte de l'application finale comporte classiquement, comme cela est représenté à la figure 3 un microprocesseur (CPU) avec tous ses registres et opérateurs internes, une  
10 mémoire morte ROM de programme, pour les programmes figés déjà mis au point et principalement pour le système d'exploitation de la carte, une mémoire vive de travail (RAM), volatile, et une mémoire non volatile programmable électriquement, de préférence EEPROM,  
15 pouvant contenir des données mais aussi du code exécutable, notamment le programme d'application qu'on cherche à mettre au point. La carte peut comprendre bien sûr d'autres circuits et notamment des circuits de sécurité établissant les autorisations d'échanges entre  
20 le lecteur et la carte. Ces échanges se font par des connexions d'entrée/sortie E/S, en principe en mode série.

Lors de la mise au point du programme d'application contenu en mémoire non volatile, on utilise une première  
25 zone Z1 de cette mémoire pour contenir le programme à déboguer, et une autre zone de mémoire Z2 pour contenir un programme d'aide au débogage. Ces deux zones contiennent du code exécutable par le microprocesseur de la carte. Une troisième zone Z3 sert de mémoire de  
30 données non volatiles; ces trois zones sont accessibles en lecture, écriture et effacement par le microprocesseur. D'autres zones peuvent encore être prévues, par exemple pour contenir des informations non accessibles en lecture et/ou en programmation et/ou en effacement par le microprocesseur.

La figure 4 représente cette organisation de la mémoire non volatile de la carte échantillon qui sert au débogage.

Le programme d'aide au débogage qui est contenu  
5 dans la zone Z2 de la mémoire est un programme  
permettant de faire exécuter par le microprocesseur  
diverses opérations permettant d'analyser le déroulement  
du programme principal; par exemple, ce programme d'aide  
comprend des instructions permettant de stocker en  
10 mémoire non volatile les contenus de registres et de la  
mémoire vive du microprocesseur, puis de les lire pour  
les envoyer au microordinateur. Il permet aussi de lire  
et modifier des zones de la mémoire vive ou de la  
mémoire non volatile.

15 La mise au point du programme se déroule de la  
manière suivante : par le microordinateur 34 et par  
l'intermédiaire du lecteur de carte 32 commandé par le  
microordinateur, on charge dans la zone Z1 de la mémoire  
non volatile de la carte le programme d'application; on  
20 modifie par exemple une instruction de ce programme pour  
y introduire un point d'arrêt (permettant d'examiner  
l'état de la carte à l'instant où ce point est atteint);  
le point d'arrêt consiste en une instruction de  
branchement vers le programme d'aide au débogage qu'on a  
25 stocké dans la zone Z2.

On fait exécuter le programme d'application jusqu'à  
l'instruction de branchement, à partir de laquelle le  
programme d'aide au débogage prend le relais pour  
effectuer certaines opérations, et notamment le relevé  
30 de certains registres et mémoires du circuit. Ces  
relevés sont stockés en mémoire EEPROM, dans la zone de  
données Z3. Ces relevés seront lus ultérieurement par le  
microordinateur 34; ils permettent de contrôler le  
fonctionnement de la carte et de détecter des erreurs de  
fonctionnement.

Le programme d'aide au débogage permet aussi de modifier le contenu de certaines zones de mémoire vive ou de la mémoire non volatile, et même le contenu de certains registres internes du microprocesseur.

5 Dans la pratique, les normes d'utilisation des cartes à puces prévoient un protocole d'échanges assez strict entre la carte et le lecteur.

Ce protocole défini par la norme ISO 7816-3 est encore appelé T = 0 ; il est représenté schématiquement à la figure 5 : c'est le lecteur de carte qui a le contrôle permanent des échanges et non la carte (pendant la mise au point de l'application, le lecteur est contrôlé par le microordinateur 34); le lecteur envoie des octets de commande CMD pour définir les opérations à effectuer par la carte; la carte répond par un octet de procédure PB; puis les échanges de données ont lieu (signaux DATA IN ou DATA OUT selon que la carte reçoit ou émet des données); et enfin la carte émet deux octets de fin de procédure ME1, ME2. Si le déroulement de la séquence n'est pas conforme à cette procédure, le lecteur envoie un message d'erreur et reprend le contrôle. Cette procédure normalisée est donnée à titre d'exemple.

Une des fonctions importantes du débogage est la connaissance du contexte machine (contenu de la mémoire vive, contenu des registres microprocesseur) à un instant donné du déroulement du programme d'application; cela veut dire qu'une séquence d'échanges entre le lecteur et la carte sera forcément interrompue si on a besoin de connaître le contexte au milieu de cette séquence. Or le lecteur réagit à toute rupture de séquence en émettant un message d'erreur et en réinitialisant tous les registres.

Selon l'invention, on propose un moyen pour permettre quand même la connaissance du contexte machine

à un instant désiré d'une séquence d'échanges.

Pour atteindre ce but, le programme d'aide au débogage (vers lequel on dérouté le programme d'application à l'endroit désiré) contient des  
5 instructions de sauvegarde des contenus registres et mémoires qu'on veut observer, et des instructions de simulation d'échanges de données avec le lecteur. La sauvegarde est faite dans la zone de données Z3 de la mémoire EEPROM.

10 Par conséquent, la séquence d'analyse se déroule de la manière suivante : le programme d'application est chargé dans la mémoire EEPROM (zone Z1) avec une instruction modifiée à l'endroit où on veut observer le contexte machine; l'instruction modifiée est une  
15 instruction de branchement vers le programme d'aide au débogage (zone Z2); le programme d'aide au débogage écrit alors en mémoire EEPROM (zone Z3) le contenu des registres et mémoires et il exécute une séquence fictive d'échanges avec le lecteur de carte (à partir du point  
20 où cette séquence a été interrompue) pour laisser croire au lecteur que la séquence en cours se déroule normalement.

Par exemple, si la séquence en cours est une séquence d'envoi de données par la carte et si on veut  
25 observer le contexte machine juste après l'octet de procédure PB, le programme d'aide au débogage enverra des données quelconques au lecteur pendant la sauvegarde du contexte machine. Si la séquence en cours était une séquence de réception de données par la carte, le  
30 programme d'aide au débogage simulerait une réception de données.

Le contexte machine sauvegardé en mémoire non volatile à des adresses spécifiées de la zone Z3 pourra être lu ultérieurement : le microordinateur 34 peut lire ou écrire à n'importe quelle adresse de la mémoire non

volatile sauf éventuellement dans des zones à accès réservé. La révélation du contenu des registres du microprocesseur par la lecture ultérieure de la zone Z3 permet de comprendre les erreurs du programme de l'application et de les corriger.

Par exemple, pour le microprocesseur ST8 de SGS-THOMSON MICROELECTRONICS S.A., la série d'instructions microprocesseurs de débogage qui servent à la sauvegarde du contenu des registres, est, en langage assembleur, la suivante :

Input Filename : BPHGR.asm  
Output Filename : BPHGR.obj

```

55      *****
56      * TSA: TRANSFERT SP TO A *
57      *****
58      TSA MACRO
15 59      FCB $9E
60      ENDM
61
62      *****
63      E600      ORG $E600
64      E600      BACKUP      BLKB 40,D      * ZONE DE SAUVEGARDE DE LA RAM ET DES
65                                     * REGISTRES A, X, CC ET SP
66
67      E628      24 04      CSET      BHS      CCLR      *
68      E62A      10 0A      BSET      0,REG_CC      *
20 69      E62C      20 02      BRA      ZSET      * RECUPEATION DE CC,A ET X
70      E62E      11 8A      CCLR      BCLR      0,REG_CC      *
71                                     *
72      E630      26 04      ZSET      BHE      ZCLR      *
73      E632      12 8A      BSET      1,REG_CC      *
74      E634      20 02      BRA      HSET      *
75      E636      13 8A      ZCLR      BCLR      1,REG_CC      *
76                                     *
25 77      E638      2A 04      NSET      BPL      NCLR      *
78      E63A      14 8A      BSET      2,REG_CC      *
79      E63C      20 02      BRA      ISET      *
80      E63E      15 8A      NCLR      BCLR      2,REG_CC      *
81                                     *
82      E640      2C 04      ISET      BHC      ICLR      *
83      E642      16 8A      BSET      3,REG_CC      *
84      E644      20 02      BRA      HSET      *
85      E646      17 8A      ICLR      BCLR      3,REG_CC      *
86                                     *
87      E648      28 04      HSET      BHCC      HCLR      *
88      E64A      18 8A      BSET      4,REG_CC      *
89      E64C      20 02      BRA      NXT_REG      *
30 90      E64E      19 8A      HCLR      BCLR      4,REG_CC      *
91                                     *
92      E650      B7 88      NXT_REG      STA      REG_A      *
93      E652      B7 89      STA      REG_X      *
94      E654      TSA      *
95      E654      9E      FCB      $9E
96      E655      ENDM
97      E655      B7 8B      STA      REG_SP      *

```

Cette série d'instructions est utilisable telle quelle sur tous les microprocesseurs du type 6805. Elle doit être modifiée en fonction du langage utilisé par le microprocesseur, pour les autres microprocesseurs. Sa structure et son but se déduisent de ceux indiqués ci-dessus.

## R E V E N D I C A T I O N S

1. Système de débogage de programme d'application de carte à mémoire comprenant une carte à mémoire échantillon (30) correspondant à celle utilisée dans l'application à déboguer, cette carte comportant au moins un microprocesseur et une mémoire non volatile programmable électriquement, un lecteur de carte (32) dans lequel cette carte est introduite, un microordinateur (34) pour contrôler le lecteur et notamment les échanges de données entre la carte et le lecteur, le programme à déboguer étant contenu dans une première zone (Z1) de la mémoire non volatile de la carte, et un programme d'aide au débogage étant contenu dans une deuxième zone (Z2) de la mémoire non volatile.

2. Système selon la revendication 1, caractérisé en ce que le programme d'aide au débogage comporte des moyens pour sauvegarder dans la mémoire non volatile des données internes représentant le fonctionnement de la carte.

3. Système selon l'une des revendications 1 et 2, caractérisé en ce qu'il comporte des moyens pour charger dans la mémoire non volatile un programme d'application comportant une instruction modifiée à l'endroit où on veut observer le comportement interne de la carte, cette instruction modifiée étant une instruction de branchement vers le programme d'aide au débogage.

4. Procédé de débogage d'application de carte à puce, ce procédé étant caractérisé en ce qu'il comprend les opérations consistant à :

- introduire une carte échantillon (30) dans un lecteur de cartes (32) contrôlé par un microordinateur (34), la carte comportant un



microprocesseur et une mémoire non volatile programmable électriquement,

5           - charger dans la mémoire non volatile, à l'aide du microordinateur et du lecteur, un programme d'application à déboguer, et un programme d'aide au débogage,

10           - donner par l'intermédiaire du microordinateur et du lecteur des ordres de modification et/ou d'exécution du programme chargé, l'exécution étant éventuellement modifiée sous le contrôle du programme d'aide au débogage, et recueillir dans le microordinateur les données résultant de l'exécution du programme,

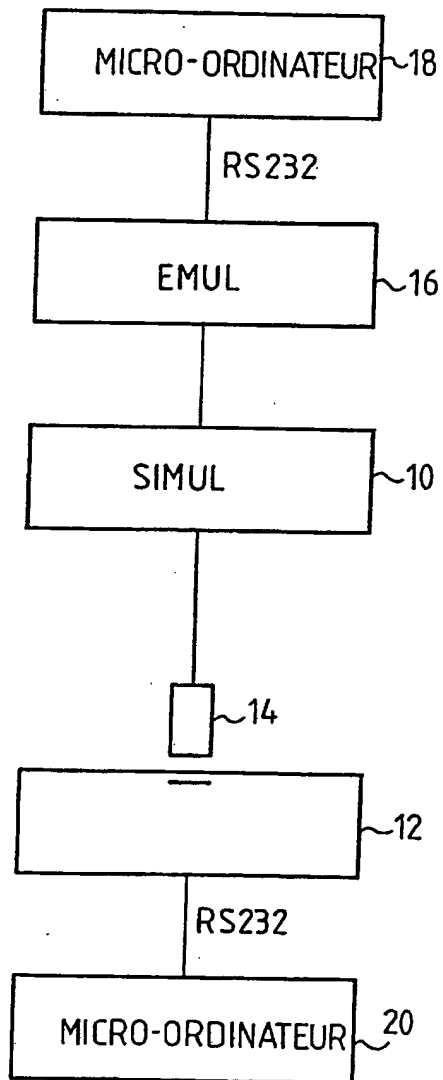
15           - modifier le programme d'application s'il y a lieu en fonction des résultats et recommencer les étapes de chargement d'un programme d'application, d'exécution, et de recueil des données.

20           5. Procédé selon la revendication 4, caractérisé en ce que le programme d'aide au débogage comporte des moyens pour sauvegarder dans la mémoire non volatile les contenus de mémoires et registres internes représentant le contexte machine du microprocesseur.

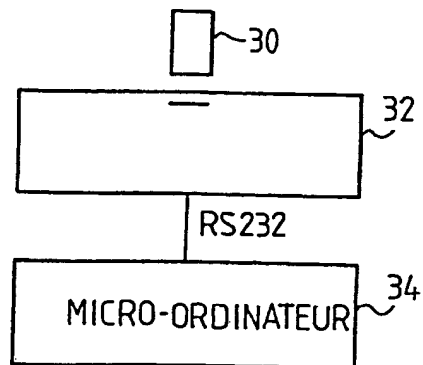
25           6. Procédé selon la revendication 5, caractérisé en ce que le programme d'aide au débogage comporte des moyens pour simuler la suite d'une séquence d'échanges entre la carte et le microprocesseur lorsqu'une telle séquence a été interrompue en vue de la sauvegarde du contexte machine dans la mémoire non volatile.

1/2

FIG\_1

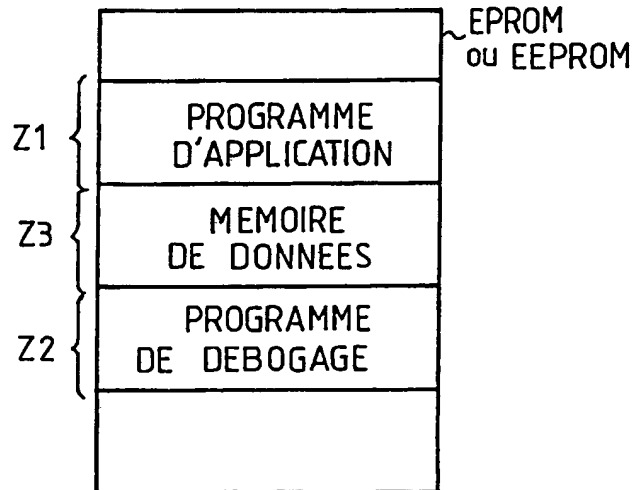
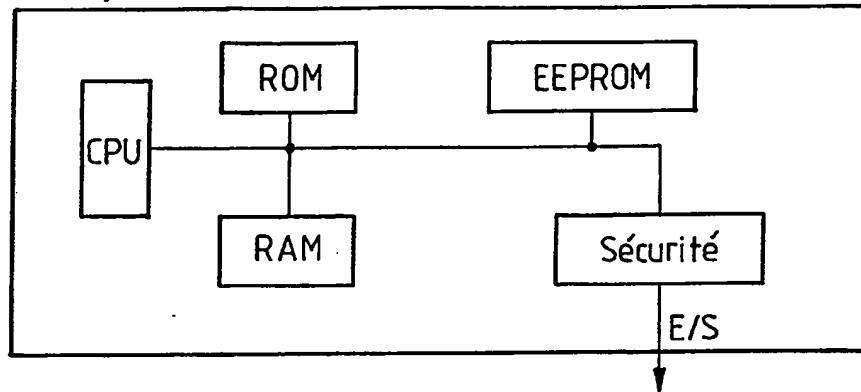


FIG\_2



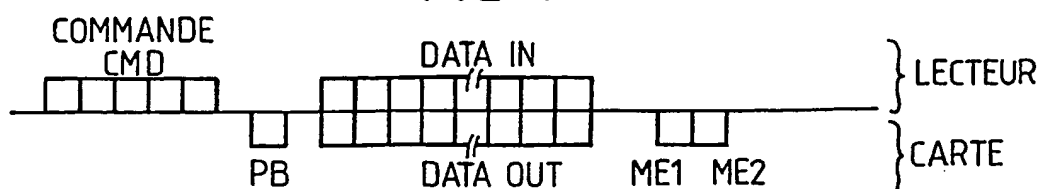
2/2

FIG\_3



FIG\_4

FIG\_5



**2667419**

Nº de enregistrement  
national

**INSTITUT NATIONAL**  
**de la**  
**PROPRIETE INDUSTRIELLE**

## RAPPORT DE RECHERCHE

**établi sur la base des dernières revendications  
déposées avant le commencement de la recherche**

FR 9012114  
FA 448145

DOCUMENTS CONSIDERES COMME PERTINENTS		Revendications concernées de la demande examinée
Catégorie	Citation du document avec indication, en cas de besoin, des parties pertinentes	
X	EP-A-0 356 237 (HITACHI) * Abrégé; colonne 2, lignes 30-65; colonne 3, lignes 1-15: colonne 3, lignes 1-57 * ---	1-6
A	US-A-4 777 355 (MITSUBISHI) * Abrégé * ---	1,2
A	FR-A-2 633 755 (MITSUBISHI) * Abrégé * -----	1,3
		DOMAINES TECHNIQUES RECHERCHES (Int. Cl.5)
		G 06 K G 07 F
Date d'achèvement de la recherche 03-05-1991		Examineur CHIARIZIA S.J
<p><b>CATEGORIE DES DOCUMENTS CITES</b></p> <p>X : particulièrement pertinent à lui seul  Y : particulièrement pertinent en combinaison avec un autre document de la même catégorie  A : pertinent à l'encontre d'au moins une revendication ou arrière-plan technologique général  O : divulgation non-écrite  P : document intercalaire</p> <p>T : théorie ou principe à la base de l'invention  E : document de brevet bénéficiant d'une date antérieure à la date de dépôt et qui n'a été publié qu'à cette date de dépôt ou qu'à une date postérieure.  D : cité dans la demande  L : cité pour d'autres raisons</p> <p>&amp; : membre de la même famille, document correspondant</p>		